Microsoft

Exam Questions DP-420

Designing and Implementing Cloud-Native Applications Using Microsoft Azure Cosmos DB

About Exambible

Your Partner of IT Exam

Found in 1998

Exambible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, Exambible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

  All examinations will be up to date.

* 24/7 Quality Support

  We will provide service round the clock.

* 100% Pass Rate

  Our guarantee that you will pass the exam.

* Unique Gurantee

  If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1
- (Exam Topic 1)
You need to select the partition key for con-iot1. The solution must meet the IoT telemetry requirements. What should you select?

A. the timestamp
B. the humidity
C. the temperature
D. the device ID

Answer: D

Explanation:
The partition key is what will determine how data is routed in the various partitions by Cosmos DB and needs to make sense in the context of your specific scenario. The IoT Device ID is generally the "natural" partition key for IoT applications.
Scenario: The iotdb database will contain two containers named con-iot1 and con-iot2. Ensure that Azure Cosmos DB costs for IoT-related processing are predictable. Reference:
https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/iot-using-cosmos-db

NEW QUESTION 2
- (Exam Topic 1)
You need to identify which connectivity mode to use when implementing App2. The solution must support the planned changes and meet the business requirements.
Which connectivity mode should you identify?

A. Direct mode over HTTPS
B. Gateway mode (using HTTPS)
C. Direct mode over TCP

Answer: C

Explanation:
Scenario: Develop an app named App2 that will run from the retail stores and query the data in account2. App2 must be limited to a single DNS endpoint when accessing account2.
By using Azure Private Link, you can connect to an Azure Cosmos account via a private endpoint. The private endpoint is a set of private IP addresses in a subnet within your virtual network.
When you're using Private Link with an Azure Cosmos account through a direct mode connection, you can use only the TCP protocol. The HTTP protocol is not currently supported.
Reference:
https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-configure-private-endpoints

NEW QUESTION 3
- (Exam Topic 2)
You maintain a relational database for a book publisher. The database contains the following tables.

Name	Column
Author	authorId **(primary key)**
	fullname
	addresse
	contactinfo
Book	bookId **(primary key)**
	isbn
	title
	genre
BookauthorInk	authorId **(foreign key)**
	bookId **(foreign key)**

The most common query lists the books for a given authorId.
You need to develop a non-relational data model for Azure Cosmos DB Core (SQL) API that will replace the relational database. The solution must minimize latency and read operation costs.
What should you include in the solution?

A. Create a container for Author and a container for Boo
B. In each Author document, embed bookId for each book by the autho
C. In each Book document embed authorId of each author.
D. Create Author, Book, and BookauthorInk documents in the same container.
E. Create a container that contains a document for each Author and a document for each Boo
F. In each Book document, embed authorId.
G. Create a container for Author and a container for Boo
H. In each Author document and Book document embed the data from BookauthorInk.

Answer: A

Explanation:
Store multiple entity types in the same container.

NEW QUESTION 4
- (Exam Topic 2)
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You have an Azure Cosmos DB Core (SQL) API account named account 1 that uses autoscale throughput. You need to run an Azure function when the normalized request units per second for a container in account1 exceeds a specific value.
Solution: You configure the function to have an Azure CosmosDB trigger. Does this meet the goal?

A. Yes
B. No

Answer: B

Explanation:
Instead configure an Azure Monitor alert to trigger the function.
You can set up alerts from the Azure Cosmos DB pane or the Azure Monitor service in the Azure portal. Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/create-alerts

NEW QUESTION 5
- (Exam Topic 2)
The following is a sample of a document in orders.

```
{
    "orderId" : "d4a91979b-5ead-43a3-b851-add9a71ac4b6",
    "customerId" : "f6e39103-bdc7-4346-9cfb-45daa4b2becf",
    "orderDate" : "2021-09-29",
 "orderItems" : [
   {
    "itemId" : "6c30412f-3cd7-4cab-813c-05942345720d",
    "name" : "blue pen",
    "type" : "pens",
    "count" : 10,
   },
   ...
 ],
 "total" : 12345,
 "status" : "ordered"
}
```

The orders container uses customerId as the partition key.
You need to provide a report of the total items ordered per month by item type. The solution must meet the following requirements:
Ensure that the report can run as quickly as possible. Minimize the consumption of request units (RUs). What should you do?

A. Configure the report to query orders by using a SQL query.
B. Configure the report to query a new aggregate containe
C. Populate the aggregates by using the change feed.
D. Configure the report to query orders by using a SQL query through a dedicated gateway.
E. Configure the report to query a new aggregate containe
F. Populate the aggregates by using SQL queries that run daily.

Answer: B

Explanation:
You can facilitate aggregate data by using Change Feed and Azure Functions, and then use it for reporting.
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/change-feed

NEW QUESTION 6
- (Exam Topic 2)
You have an Azure Cosmos DB Core (SQL) API account.
You configure the diagnostic settings to send all log information to a Log Analytics workspace.
You need to identify when the provisioned request units per second (RU/s) for resources within the account were modified.
You write the following query. AzureDiagnostics
| where Category == "ControlPlaneRequests" What should you include in the query?

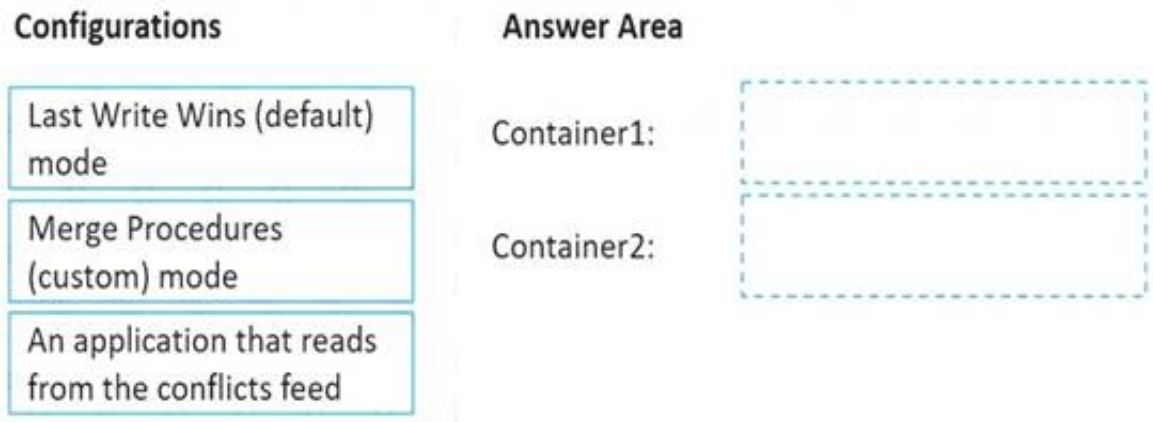
A. | where OperationName startswith "AccountUpdateStart"
B. | where OperationName startswith "SqlContainersDelete"
C. | where OperationName startswith "MongoCollectionsThroughputUpdate"
D. | where OperationName startswith "SqlContainersThroughputUpdate"

Answer: A

Explanation:
The following are the operation names in diagnostic logs for different operations: RegionAddStart, RegionAddComplete RegionRemoveStart, RegionRemoveComplete AccountDeleteStart, AccountDeleteComplete RegionFailoverStart, RegionFailoverComplete AccountCreateStart,

AccountCreateComplete
AccountUpdateStart, AccountUpdateComplete VirtualNetworkDeleteStart, VirtualNetworkDeleteComplete DiagnosticLogUpdateStart, DiagnosticLogUpdateComplete
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/audit-control-plane-logs

NEW QUESTION 7
- (Exam Topic 2)
You have an Azure Cosmos DB Core (SQL) API account that is configured for multi-region writes. The account contains a database that has two containers named container1 and container2.
The following is a sample of a document in container1:
{
"customerId": 1234, "firstName": "John",
"lastName": "Smith", "policyYear": 2021
}
The following is a sample of a document in container2:
{
"gpsId": 1234,
"latitude": 38.8951,
"longitude": -77.0364
}
You need to configure conflict resolution to meet the following requirements:
For container1 you must resolve conflicts by using the highest value for policyYear.
For container2 you must resolve conflicts by accepting the distance closest to latitude: 40.730610 and longitude: -73.935242.
Administrative effort must be minimized to implement the solution.
What should you configure for each container? To answer, drag the appropriate configurations to the correct containers. Each configuration may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.
NOTE: Each correct selection is worth one point.



A. Mastered
B. Not Mastered

Answer: A

Explanation:
Box 1: Last Write Wins (LWW) (default) mode
Last Write Wins (LWW): This resolution policy, by default, uses a system-defined timestamp property. It's based on the time-synchronization clock protocol.
Box 2: Merge Procedures (custom) mode
Custom: This resolution policy is designed for application-defined semantics for reconciliation of conflicts. When you set this policy on your Azure Cosmos container, you also need to register a merge stored procedure. This procedure is automatically invoked when conflicts are detected under a database transaction on the server. The system provides exactly once guarantee for the execution of a merge procedure as part of the commitment protocol.
Reference:
https://docs.microsoft.com/en-us/azure/cosmos-db/conflict-resolution-policies https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-manage-conflicts

NEW QUESTION 8
- (Exam Topic 2)
You have three containers in an Azure Cosmos DB Core (SQL) API account as shown in the following table.

Name	Database	Time to Live
cn1	db1	On (no default)
cn2	db1	Off
cn3	db1	On (no default)

You have the following Azure functions:
A function named Fn1 that reads the change feed of cn1 A function named Fn2 that reads the change feed of cn2 A function named Fn3 that reads the change feed of cn3
You perform the following actions: Delete an item named item1 from cn1. Update an item named item2 in cn2.
For an item named item3 in cn3, update the item time to live to 3,600 seconds.
For each of the following statements, select Yes if the statement is true. Otherwise, select No. NOTE: Each correct selection is worth one point.

Answer Area

Statements	Yes	No
Fn1 will receive item1 from the change feed	○	○
Fn2 can check the _etag of item2 to see whether the item is an update or an insert	○	○
Fn3 will receive item3 from the change feed	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Box 1: No
Azure Cosmos DB's change feed is a great choice as a central data store in event sourcing architectures where all data ingestion is modeled as writes (no updates or deletes).
Note: The change feed does not capture deletes. If you delete an item from your container, it is also removed from the change feed. The most common method of handling this is adding a soft marker on the items that are being deleted. You can add a property called "deleted" and set it to "true" at the time of deletion. This document update will show up in the change feed. You can set a TTL on this item so that it can be automatically deleted later.
Box 2: No
The _etag format is internal and you should not take dependency on it, because it can change anytime.
Box 3: Yes
Change feed support in Azure Cosmos DB works by listening to an Azure Cosmos container for any changes. Reference:
https://docs.microsoft.com/en-us/azure/cosmos-db/sql/change-feed-design-patterns https://docs.microsoft.com/en-us/azure/cosmos-db/change-feed

NEW QUESTION 9
- (Exam Topic 2)
You have a container in an Azure Cosmos DB Core (SQL) API account.
You need to use the Azure Cosmos DB SDK to replace a document by using optimistic concurrency. What should you include in the code? To answer, select the appropriate options in the answer area. NOTE: Each correct selection is worth one point.

Answer Area

RequestOptions property to set: [▼]
- AccessCondition
- ConsistencyLevel
- SessionToken

Document property that will be compared: [▼]
- _etag
- _id
- _rid

A. Mastered
B. Not Mastered

Answer: A

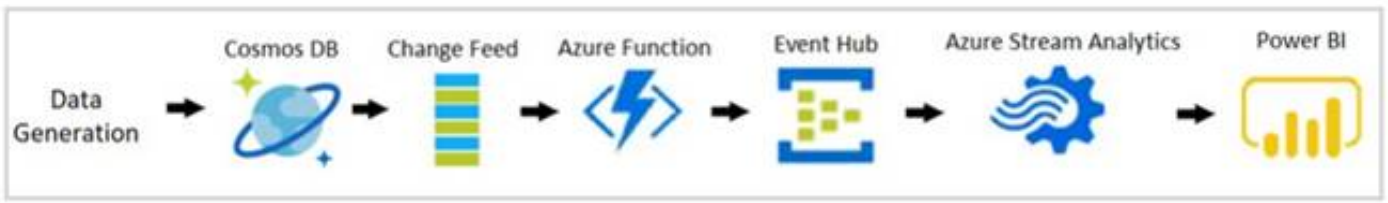
Explanation:
Box 1: ConsistencyLevel
The ItemRequestOptions Class ConsistencyLevel property gets or sets the consistency level required for the request in the Azure Cosmos DB service.
Azure Cosmos DB offers 5 different consistency levels. Strong, Bounded Staleness, Session, Consistent Prefix and Eventual - in order of strongest to weakest consistency.
Box 2: _etag
The ItemRequestOptions class helped us implement optimistic concurrency by specifying that we wanted the SDK to use the If-Match header to allow the server to decide whether a resource should be updated. The If-Match value is the ETag value to be checked against. If the ETag value matches the server ETag value, the resource is updated.
Reference:
https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.cosmos.itemrequestoptions https://cosmosdb.github.io/labs/dotnet/labs/10-concurrency-control.html

NEW QUESTION 10
- (Exam Topic 2)
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You have a container named container1 in an Azure Cosmos DB Core (SQL) API account.
You need to make the contents of container1 available as reference data for an Azure Stream Analytics job. Solution: You create an Azure Data Factory pipeline that uses Azure Cosmos DB Core (SQL) API as the input and Azure Blob Storage as the output.
Does this meet the goal?

A. Yes
B. No

Answer: B

Explanation:
Instead create an Azure function that uses Azure Cosmos DB Core (SQL) API change feed as a trigger and Azure event hub as the output.
The Azure Cosmos DB change feed is a mechanism to get a continuous and incremental feed of records from an Azure Cosmos container as those records are being created or modified. Change feed support works by listening to container for any changes. It then outputs the sorted list of documents that were changed in the order in which they were modified.
The following diagram represents the data flow and components involved in the solution:



C:\Users\Admin\Desktop\Data\Odt data\Untitled.jpg
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/sql/changefeed-ecommerce-solution

NEW QUESTION 10
- (Exam Topic 2)
You are developing an application that will use an Azure Cosmos DB Core (SQL) API account as a data source.
You need to create a report that displays the top five most ordered fruits as shown in the following table.

Name	Type	Orders
apple	fruit	1,000
orange	fruit	600
banana	fruit, exotic	400
plum	fruit.	300
mango	fruit, exotic	200

A collection that contains aggregated data already exists. The following is a sample document:
{
"name": "apple",
"type": ["fruit", "exotic"], "orders": 10000
}
Which two queries can you use to retrieve data for the report? Each correct answer presents a complete solution.
NOTE: Each correct selection is worth one point.
A)

```
SELECT TOP  i.name, i.types, i.orders
FROM items i
WHERE EXISTS(SELECT VALUE t FROM t IN i.types WHERE t.name = 'fruit')
ORDER BY i.orders,i.types
```

B)

```
SELECT TOP  i.name, i.types, i.orders
FROM items i
WHERE EXISTS(SELECT VALUE t FROM t IN i.types WHERE t.name = 'fruit')
ORDER BY i.orders DESC
```

C)

```
SELECT TOP  i.name, i.types, i.orders
FROM items i
WHERE EXISTS(SELECT VALUE t FROM t IN i.types WHERE t.name = 'fruit')
ORDER BY i.types DESC
```

D)

```
SELECT TOP  i.name, i.types, i.orders
FROM items i
WHERE ARRAY_CONTAINS(i.types, {name: 'fruit'})
ORDER BY i.orders DESC
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: BD

Explanation:

ARRAY_CONTAINS returns a Boolean indicating whether the array contains the specified value. You can check for a partial or full match of an object by using a boolean expression within the command.
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/sql/sql-query-array-contains

NEW QUESTION 14
- (Exam Topic 2)
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You have an Azure Cosmos DB Core (SQL) API account named account 1 that uses autoscale throughput. You need to run an Azure function when the normalized request units per second for a container in account1 exceeds a specific value.
Solution: You configure an application to use the change feed processor to read the change feed and you configure the application to trigger the function.
Does this meet the goal?

A. Yes
B. No

Answer: B

Explanation:
Instead configure an Azure Monitor alert to trigger the function.
You can set up alerts from the Azure Cosmos DB pane or the Azure Monitor service in the Azure portal. Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/create-alerts

NEW QUESTION 19
- (Exam Topic 2)
You have an Azure Cosmos DB Core (SQL) API account named account1 that has the disableKeyBasedMetadataWriteAccess property enabled.
You are developing an app named App1 that will be used by a user named DevUser1 to create containers in account1. DevUser1 has a non-privileged user account in the Azure Active Directory (Azure AD) tenant.
You need to ensure that DevUser1 can use App1 to create containers in account1. What should you do? To answer, select the appropriate options in the answer area. NOTE: Each correct selection is worth one point.

Answer Area

Grant permissions to create containers by using: [▼]
- Account keys
- Resource tokens
- Role-based access control (RBAC)

Create containers by using the: [▼]
- Azure AD Graph API
- Azure Resource Manager API
- SQL (Core) API

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Box 1: Resource tokens
Resource tokens provide access to the application resources within a database. Resource tokens:
Provide access to specific containers, partition keys, documents, attachments, stored procedures, triggers, and UDFs.
Box 2: Azure Resource Manager API
You can use Azure Resource Manager to help deploy and manage your Azure Cosmos DB accounts, databases, and containers.
Reference:
https://docs.microsoft.com/en-us/azure/cosmos-db/secure-access-to-data https://docs.microsoft.com/en-us/rest/api/resources/

NEW QUESTION 20
- (Exam Topic 2)
You are designing an Azure Cosmos DB Core (SQL) API solution to store data from IoT devices. Writes from the devices will be occur every second.
The following is a sample of the data.

```
{
    "id" : "03c1ca5a-db18-4231-908f-09a9bc7a7c3e",
    "deviceManufacturer" : "Contoso, Ltd",
    "deviceId" : "f460df85-799f-4d58-b051-67561b4993c6",
    "timestamp" : "2021-09-19T13:47:45",
    "sensor1Value" : true,
    "sensor2Value" : "75",
    "sensor3Value" : "4554",
    "sensor4Value" : "454",
    "sensor5Value" : "42128"
}
```

You need to select a partition key that meets the following requirements for writes: Minimizes the partition skew
Avoids capacity limits Avoids hot partitions What should you do?

A. Use timestamp as the partition key.
B. Create a new synthetic key that contains deviceId and sensor1Value.
C. Create a new synthetic key that contains deviceId and deviceManufacturer.
D. Create a new synthetic key that contains deviceId and a random number.

Answer: D

Explanation:
Use a partition key with a random suffix. Distribute the workload more evenly is to append a random number at the end of the partition key value. When you distribute items in this way, you can perform parallel write operations across partitions.
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/sql/synthetic-partition-keys

NEW QUESTION 24
- (Exam Topic 2)
You have an Azure Cosmos DB Core (SQL) account that has a single write region in West Europe. You run the following Azure CLI script.

```
az cosmosdb update -n $accountName -g $resourceGroupName \
   --locations regionName='West Europe' failoverPriority=0 isZoneRedundant-False \
   --locations regionName='North Europe' failoverPriority=1 isZoneRedundant-False

az cosmosdb failover-priority-change -n $accountName -g $resourceGroupName \
   --failover-policies 'North Europe=0' 'West Europe=1'
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No. NOTE: Each correct selection is worth one point.

Answer Area

Statements	Yes	No
After running the script, there will be an instance of Azure Cosmos DB in North Europe that is writable	○	○
After running the script, the Azure Cosmos DB instance in West Europe will be writable	○	○
The cost of the Azure Cosmos DB account is unaffected by running the script	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Box 1: Yes
The Automatic failover option allows Azure Cosmos DB to failover to the region with the highest failover priority with no user action should a region become unavailable.
Box 2: No
West Europe is used for failover. Only North Europe is writable. To Configure multi-region set UseMultipleWriteLocations to true.
Box 3: Yes
Provisioned throughput with single write region costs $0.008/hour per 100 RU/s and provisioned throughput with multiple writable regions costs $0.016/per hour per 100 RU/s.
Reference:
https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-multi-master https://docs.microsoft.com/en-us/azure/cosmos-db/optimize-cost-regions

NEW QUESTION 25
- (Exam Topic 2)
You are creating a database in an Azure Cosmos DB Core (SQL) API account. The database will be used by an application that will provide users with the ability to share online posts. Users will also be able to submit comments on other users' posts.
You need to store the data shown in the following table.

Type	Description
Users	Information about a user who will use the application
Posts	Text of up to 1,000 characters that a user will share with other users
Comments	Text of up to 280 characters that users will submit as a comment on a post
Interests	Information about a user's interests

The application has the following characteristics: Users can submit an unlimited number of posts.
The average number of posts submitted by a user will be more than 1,000. Posts can have an unlimited number of comments from different users.
The average number of comments per post will be 100, but many posts will exceed 1,000 comments. Users will be limited to having a maximum of 20 interests.
For each of the following statements, select Yes if the statement is true. Otherwise, select No. NOTE: Each correct selection is worth one point.

Answer Area

Statements	Yes	No
If you embed the posts data into the users data instead of creating a separate document for each post, you will increase the write operation costs for new posts	○	○
If you embed the comments data into the posts data instead of creating a separate document for each comment you will increase the write operation costs for new comments	○	○
If you embed the interests data into the users data instead of creating a separate document for each interest, you will increase the read operation costs for displaying the users and their associated interests	○	○

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Box 1: Yes
Non-relational data increases write costs, but can decrease read costs.
Box 2: Yes
Non-relational data increases write costs, but can decrease read costs.
Box 3: No
Non-relational data increases write costs, but can decrease read costs.

NEW QUESTION 27
- (Exam Topic 2)
You have a database in an Azure Cosmos DB Core (SQL) API account.
You need to create an Azure function that will access the database to retrieve records based on a variable named accountnumber. The solution must protect against SQL injection attacks.
How should you define the command statement in the function?

A. cmd = "SELECT * FROM Persons pWHERE p.accountnumber = 'accountnumber'"
B. cmd = "SELECT * FROM Persons pWHERE p.accountnumber = LIKE @accountnumber"
C. cmd = "SELECT * FROM Persons pWHERE p.accountnumber = @accountnumber"
D. cmd = "SELECT * FROM Persons pWHERE p.accountnumber = '" + accountnumber + """

Answer: C

Explanation:
Azure Cosmos DB supports queries with parameters expressed by the familiar @ notation. Parameterized SQL provides robust handling and escaping of user input, and prevents accidental exposure of data through SQL injection.
For example, you can write a query that takes lastName and address.state as parameters, and execute it for various values of lastName and address.state based on user input.
SELECT *
FROM Families f
WHERE f.lastName = @lastName AND f.address.state = @addressState
Reference: https://docs.microsoft.com/en-us/azure/cosmos-db/sql/sql-query-parameterized-queries

NEW QUESTION 31
.....

Relate Links

100% Pass Your DP-420 Exam with Exambible Prep Materials

https://www.exambible.com/DP-420-exam/

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - https://www.exambible.com/